Manuel d'installation et de configuration du serveur Oracle 10g Standard Edition

Pour la société structis, représentée par Mr Claude GROUX.
Installation, configuration et documentation par Pascal CECILLON, société Proservia.

La société Proservia est représentée par Mr Jean-Marc ALPAGO.

I . Installation

I.1 *Comptes et groupes utilisateurs système :*

Il existe certains pré-requis à l'installation d'Oracle sur plate-forme type Unix au niveau des comptes et groupes utilisateurs . Le programme OUI (Oracle Universal Installer) ne peut être exécuté que sous le compte système 'oracle'. Pour cet utilisateur, ont été définis deux groupes : 'dba' et 'oinstall'.

Le groupe dba permet aux utilisateurs faisant partie de ce groupe de se connecter à une base du serveur en tant que SYSDBA (c'est à dire en tant qu'administrateur de la base de données).

Le groupe 'oinstall' permet de protéger l'accès au répertoire contenant l'inventaire des produits Oracle installés sur la machine. Ces deux noms ont été choisis en respectant les « traditions » Oracle.

Les commandes passées pour la création des groupes et du compte oracle :

```
groupadd oinstall
groupadd dba
useradd –g oinstall –G dba –s /bin/bash oracle
```

Le compte oracle a donc comme groupe principal oinstall et comme groupe secondaire dba.

I.2 *Vérifications et modifications des paramètres du kernel (root) :*

Oracle 10g requiert que certains paramètres du kernel aient des valeurs spécifiques. Voici les paramètres et leurs valeurs telles qu'elles sont après modification :

```
kernel.sem = 250 32000 100 128
kernel.shmmax = 1073741824
fs.file-max = 65536
net.ipv4.ip_local_port_range = 1024 65000
net.core.rmem_default = 262144
net.core.rmem_max = 262144
net.core.wmem_default = 262144
net.core.wmem_max = 262144
```

Pour modifier ces valeurs, les lignes ci-dessus ont été ajoutées au fichier /etc/sysctl.conf. Les modifications ont été validées avec la commande

```
sysctl –p
```

On vérifie ensuite la version du kernel par la commande

```
uname –r
```

Le kernel doit être au minimum en version 2.6.9. Le kernel utilisé est installé en version 2.6.9.

I.3 *Vérification et installation des packages rpm (root) :*

Oracle requiert que certains packages soient installés sur la machine dans des versions minimum. Voici les packages et leur version minimum que requiert Oracle 10g :

```
make-3.79
binutils-2.14
gcc-3.2
libaio-0.3.96
```

Voici les versions qui ont été installées sur la machine :

```
make-3.80
binutils-2.15.92.0.2-10
gcc-3.4.3-9
libaio-0.3.102-1
```

par la commande

```
rpm –ivh nom_package
```

I.4 *Vérification des partitions(FileSystems) et des exigences matérielles (root) :*

Oracle requiert qu'une partition swap d'au moins 1Go et qu'un espace temporaire généralement situé en /tmp d'au minimum 400Mo soient présents sur la machine. On vérifie ces exigences par la commande :

```
df –k
```

Le serveur dispose de 2Go de swap et d'un espace temporaire largement suffisant, /tmp n'étant pas été monté en partition.

Il faut s'assurer que le serveur dispose de suffisamment d'espace disque : au moins 1,5Go pour les produits Oracle et au moins 1,2Go de plus si l'on veut créer une base de données lors de la première installation.
Le serveur dispose ici de 4 disques de 150Go à 15000tpm, le tout en RAID5 et partitionné grâce à LVM (Logical Volume Manager). Les exigences matérielles sont donc largement remplies.

De plus, il est recommandé par l'architecture OFA de créer une partition séparée qui hébergera spécifiquement les produits Oracle. La partition /ora01 de 20Go a donc été créée sur le serveur pour accueillir Oracle.

I.5 *Création des répertoires d'installation des produits Oracle (root) :*

Oracle utilise deux répertoires principaux qui doivent valoriser deux variables d'environnement respectivement nommées $ORACLE_BASE et $ORACLE_HOME. La variable $ORACLE_BASE définit la base de l'installation d'Oracle. La variable $ORACLE_HOME définit le répertoire où sont installés les produits Oracle pour une installation donnée. On utilise donc la partition /ora01 créée précédemment.

OFA recommande la nomenclature type suivante pour ces deux répertoires :

ORACLE_BASE : /x01/yyy/compte – x étant un ou plusieurs caractères quelconques, y étant une suite de 3 caractères quelconques et compte étant le compte utilisé pour l'installation.

ORACLE_HOME : $ORACLE_BASE/product/n.n.n/type_o – n.n.n étant le numéro de la version d'Oracle, type étant le type d'installation (db pour un serveur et client pour un client) et o étant le numéro d'ordre du type d'installation (dans le cas ou l'on ai plusieurs serveurs/clients sur la même machine).

Voici les deux répertoires pour Oracle qui ont été crées , leur nomenclature respectant les recommandations OFA :

```
ORACLE_BASE : /ora01/app/oracle
ORACLE_HOME : /ora01/app/oracle/product/10.2.0/db_1
```

La création du répertoire ORACLE_HOME est assurée par le programme d'installation OUI en se basant sur le répertoire ORACLE_BASE. Il faut donc créer le répertoire ORACLE_BASE par la commande :

```
mkdir –p /ora01/app/oracle
```

Puis on accorde les droits sur ce répertoire par les commandes :

```
chown –R oracle:oinstall /ora01/app/oracle
chmod –R 775 /ora01/app/oracle
```

Concrètement, on rend oracle et le groupe oinstall propriétaires de toute l'arborescence qui constitue le chemin du répertoire ORACLE_BASE, puis on accorde les pleins-pouvoirs à cet utilisateur et à ce groupe ainsi que les droits lecture et d'exécution à tous les autres sur toute l'arborescence du répertoire ORACLE_BASE.

I.6 *Lancement du programme d'installation OUI (root / oracle) :*

On a décompressé le fichier zip téléchargé sur le site d'Oracle dans un répertoire quelconque situé sur une partition autre que /ora01 par la commande

```
unzip fichier.zip
```

Par défaut et avec le fichier téléchargé, les fichiers décompressés sont stockés dans un répertoire database situé dans le répertoire dans lequel se trouvait le fichier zip au moment de la décompression. Exemple, si le fichier zip se trouve dans /home/oracle, alors les fichiers décompressés se trouvent dans /home/oracle/database.

Il faut maintenant ouvrir une session sur l'interface graphique de la machine sous login oracle.

Sur l'interface graphique, on ouvre un nouveau terminal de commandes, dans lequel on va pouvoir lancer le fichier runInstaller se trouvant à la racine du répertoire database crée lors de la décompression du fichier zip, par la commande :

```
./runInstaller
```

Le programme d'installation refusera de se lancer si on lance le script runInstaller sous un login différent de oracle.

Si tout est bon, le programme d'installation OUI se lance, il n'y a plus qu'à suivre les instructions à l'écran que nous ne détaillerons pas ici mais qui, bien que peut-être peu claires pour un néophyte total, s'avérerons simple à exécuter si l'on a bien lu et compris ce document jusqu'ici.

II . Post-installation

II.1 *Création d'une première base de test (OUI) :*

Une base de test nommée ORATEST a été créée tout de suite après l'installation du serveur Oracle, afin de pouvoir procéder à différents essais de configuration et de paramétrages. Cette base pourra être utilisée pour dupliquer une base en production, ou bien d'autres tests de requêtage SQL etc…

La liste des bases présentes sur le serveur se trouve dans le fichier /etc/oratab.

II.2 *Configuration du lancement et de l'arrêt automatique d'Oracle (oracle) :*

Afin que les différents produits Oracle se lancent lors du boot du serveur, il y a certains paramétrages à faire au niveau du système. Oracle fourni un script de lancement automatique de toutes les bases hébergées sur le serveur ainsi que du listener. Ce fichier a été édité pour ne lpus qu'il lance le listener. Suite a des problèmes de configuration de certaines variables d'environnement, le lancement du listener a été reprogrammé manuellement. Il s'agit du script /ora01/app/oracle/product/10.2.0/db_1/bin/dbstart.

Ce script est appelé par un autre script dont nous verrons le détail plus loin et qui est le suivant : /etc/init.d/oracle.

Pour exécuter le script oracle lors du démarrage de la machine, un lien symbolique a été crée vers ce fichier oracle dans /etc/rc.d/rc5.d/ et /etc/rc.d/rc3.d/, afin que Oracle se lance lorsque le système démarre en runlevel 3 (sans interface graphique) et en runlevel 5 (boot sur interface graphique). Le lien a été crée par la commande :

```
ln –s /etc/init.d/oracle S99Oracle
```

S99 indique que l'on appelle le script oracle avec le paramètre start . 99 indique le numéro d'ordre de lancement du script oracle. Il s'exécute donc ici à la fin de la phase de boot.

Pour l'arrêt des services Oracle lors de l'extinction du serveur (en cas de shutdown ou de reboot), il a été fait la même chose que pour le lancement, mais avec le script /ora01/app/oracle/product/10.2.0/db_1/bin/dbshut fourni lui aussi par Oracle.

Le même script oracle appelle ce script en cas d'arrêt du serveur grâce au lien crée dans /etc/rc.d/rc0.d (shutdown) et dans /etc/rc.d/rc6.d (reboot) par la commande :

```
ln –s /etc/init.d/oracle K01Oracle
```

K01 indique que l'on appelle le script oracle avec le paramètre stop. 01 indique le numéro d'ordre d'arrêt du scrip oracle. Il s'exécute ici en premier lors de l'arrêt de la machine.

II.3 *Configuration des variables d'environnement et du path (root) :*

Nous avons vu au début de ce document que l'installation utilisait deux répertoires qui valorisent deux variables : $ORACLE_BASE et $ORACLE_HOME. La troisième variable $ORACLE_SID doit contenir le nom de l'instance sur laquelle on veut travailler durant la session. Elles ont donc été définies toutes les trois dans le fichier /home/root/.bash_profile par les lignes :

```
ORACLE_BASE=/ora01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/10.2.0/db_1
export ORACLE_BASE ORACLE_HOME
```

La variable $ORACLE_SID est positionnée grâce à un script qui propose à l'administrateur lors de sa connexion en terminal de choisir quelle instance doit valoriser la variable $ORACLE_SID.

Le script qui permet ce choix sera détaillé plus loin dans ce document. Ce système de positionnement n'a pu être retenu pour la connexion du DBA de la machine (login oracle) car le script de lancement automatique d'Oracle lance un su – oracle lors de son exécution, comme nous le verrons plus loin. Le script de positionnement de la variable $ORACLE_SID posant une question à l'utilisateur qui se connecte, cela bloque le démarrage de la machine, l'utilisateur ne pouvant pas disposer de terminal lors du boot du serveur.

De ce fait, sous le login oracle, il faut positionner manuellement la variable $ORACLE_SID par la commande :

```
ORACLE_SID=nom_instance
export ORACLE_SID
```

La plupart des exécutables d'Oracle se trouvent dans le répertoire $ORACLE_HOME/bin/. Afin que l'utilisateur puisse lancer les programmes qui s'y trouvent sans avoir à entrer le chemin absolu à chaque fois, le chemin $ORACLE_HOME/bin/ a été ajouté au path dans le fichier .bash_profile du répertoire $HOME des utilisateurs par la ligne :

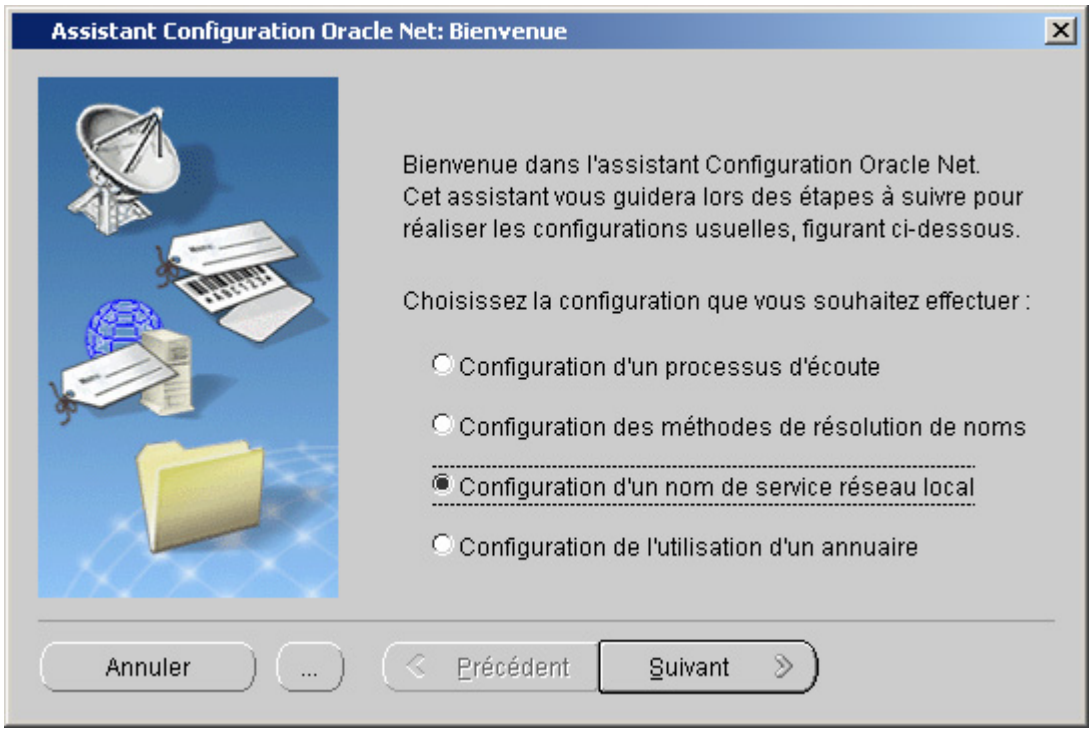
```
PATH=$PATH:/etc/init.d/:/usr/local/bin/:$ORACLE_HOME/bin/
export PATH
```

III . Configuration des clients

Dans la salle, trois postes clients Windows étaient présents lors de l'installation. Il a donc été installé sur chacun de ces postes un client Oracle. Ce client est déjà configuré pour se connecter à la base ORATEST qui a été créée lors de l'installation du serveur. Pour pouvoir se connecter à la base depuis les postes clients sur une base nouvellement créée, il est nécessaire de procéder aux différentes étapes de configuration suivantes :

Menu Démarrer > Programmes > Oracle – OraClient10g_home1 > Outils de configuration et de migration, puis sélectionner Assistant Configuration Oracle Net.

L'écran suivant s'affiche :


Assistant Configuration Oracle Net: Bienvenue
Bienvenue dans l'assistant Configuration Oracle Net.
Cet assistant vous guidera lors des étapes à suivre pour réaliser les configurations usuelles, figurant ci-dessous.
Choisissez la configuration que vous souhaitez effectuer :
Configuration d'un processus d'écoute
Configuration des méthodes de résolution de noms
Configuration d'un nom de service réseau local
Configuration de l'utilisation d'un annuaire
Annuler
...
Précédent
Suivant

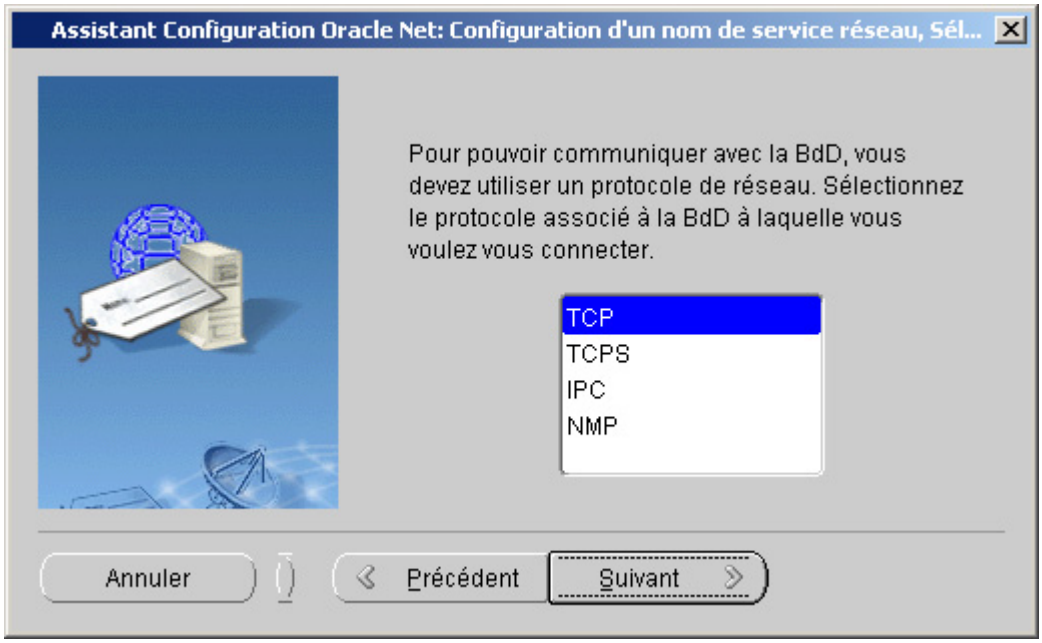
Sélectionner « Configuration d'un nom de service réseau local » et Suivant

Assistant Configuration Oracle Net: Configuration d'un nom de service réseau
Pour pouvoir accéder à une BdD Oracle ou à un autre service sur un réseau, vous devez utiliser un nom de service réseau. L'assistant Configuration Oracle Net vous permet d'utiliser des noms de service réseau résolus à l'aide de la méthode de résolution locale de noms.
Faites votre choix :
Ajouter
Reconfigurer
Supprimer
Renommer
Tester
Annuler
Aide
Précédent
Suivant

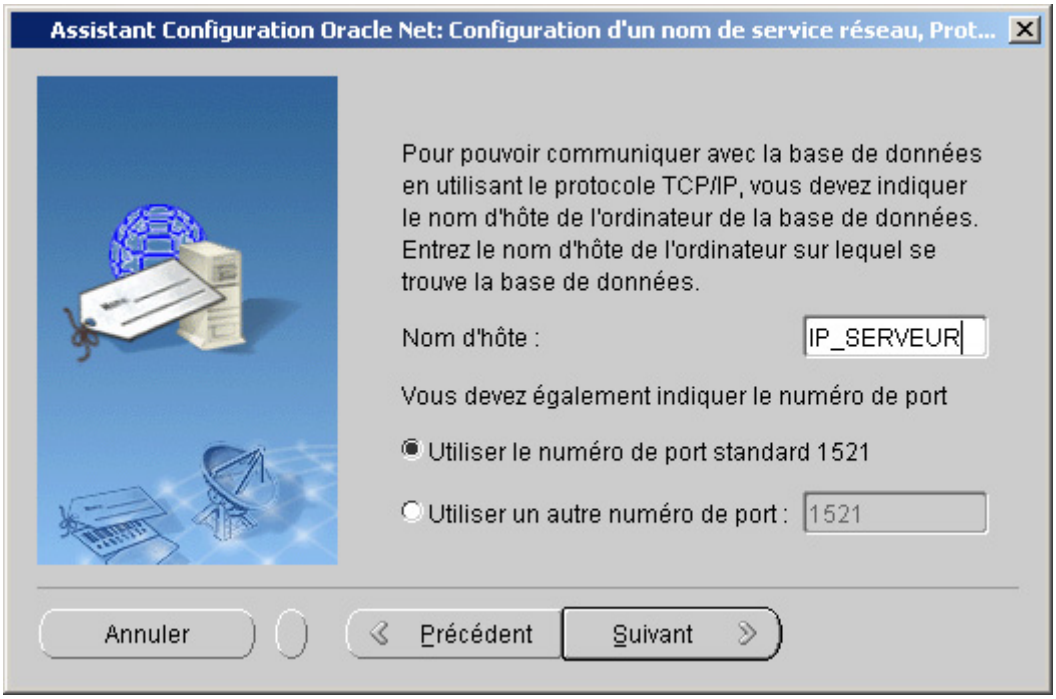
Sélectionner « Ajouter » puis Suivant

Assistant Configuration Oracle Net: Configuration d'un nom de service réseau, Nom de...
Chaque service ou base de données Oracle a un nom de service. Le nom de service d'une base Oracle est généralement son nom global de base de données. Indiquez le nom de service de la base de données ou du service auquel vous voulez accéder.
Nom de service :
NOM_NOUVELLE_BASE
Annuler
...
Précédent
Suivant

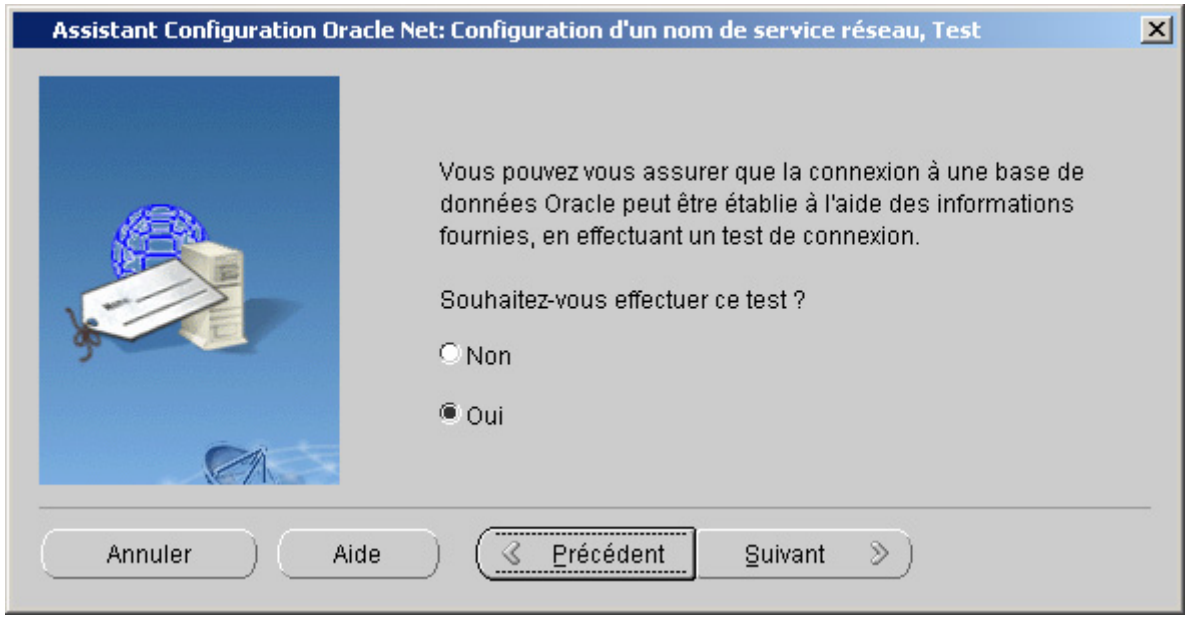
Dans « Nom de service », entrer le nom de la nouvelle base qui a été créée sur le serveur, puis Suivant.


Assistant Configuration Oracle Net: Configuration d'un nom de service réseau, Sél...
Pour pouvoir communiquer avec la BdD, vous devez utiliser un protocole de réseau. Sélectionnez le protocole associé à la BdD à laquelle vous voulez vous connecter.
TCP
TCPS
IPC
NMP
Annuler
Précédent
Suivant

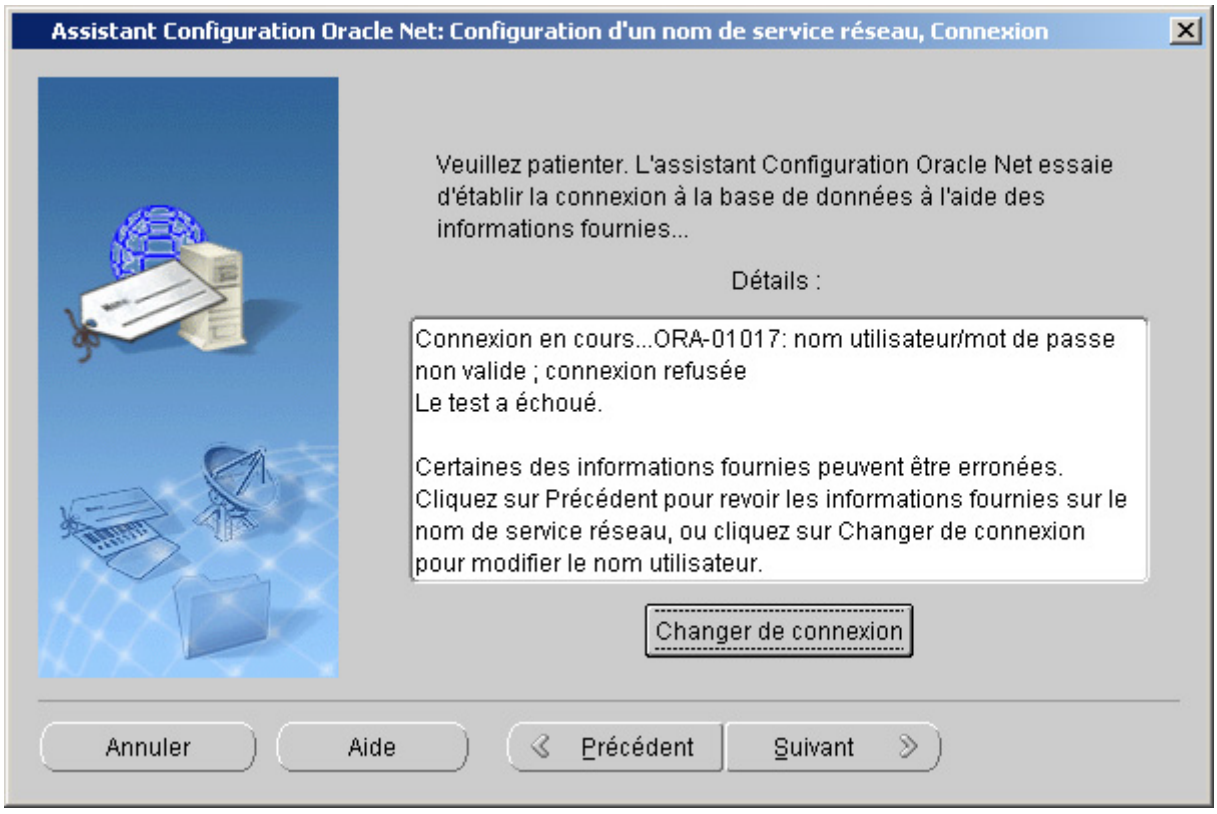
Sélectionner « TCP » puis Suivant


Assistant Configuration Oracle Net: Configuration d'un nom de service réseau, Prot...
Pour pouvoir communiquer avec la base de données en utilisant le protocole TCP/IP, vous devez indiquer le nom d'hôte de l'ordinateur de la base de données. Entrez le nom d'hôte de l'ordinateur sur lequel se trouve la base de données.
Nom d'hôte :
IP_SERVEUR
Vous devez également indiquer le numéro de port
Utiliser le numéro de port standard 1521
Utiliser un autre numéro de port :
1521
Annuler
Précédent
Suivant

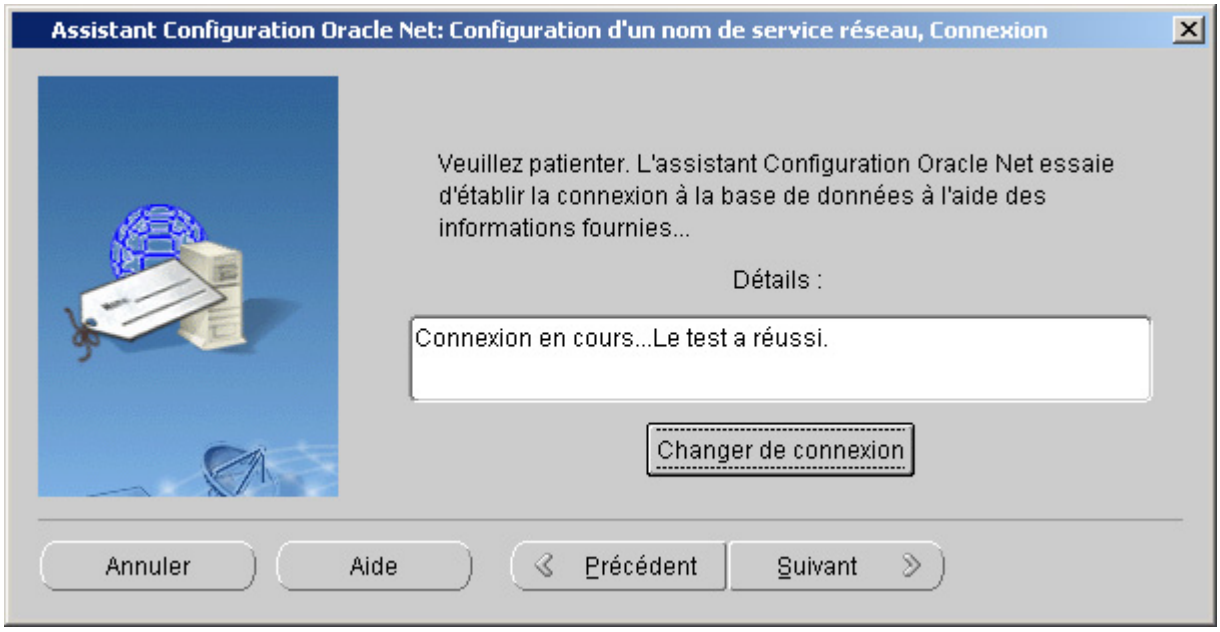
Dans « Nom d'hôte », entrer l'adresse IP du serveur, laisser le numéro de port standard, puis Suivant.


Assistant Configuration Oracle Net: Configuration d'un nom de service réseau, Test
Vous pouvez vous assurer que la connexion à une base de données Oracle peut être établie à l'aide des informations fournies, en effectuant un test de connexion.
Souhaitez-vous effectuer ce test ?
Non
Oui
Annuler
Aide
Précédent
Suivant

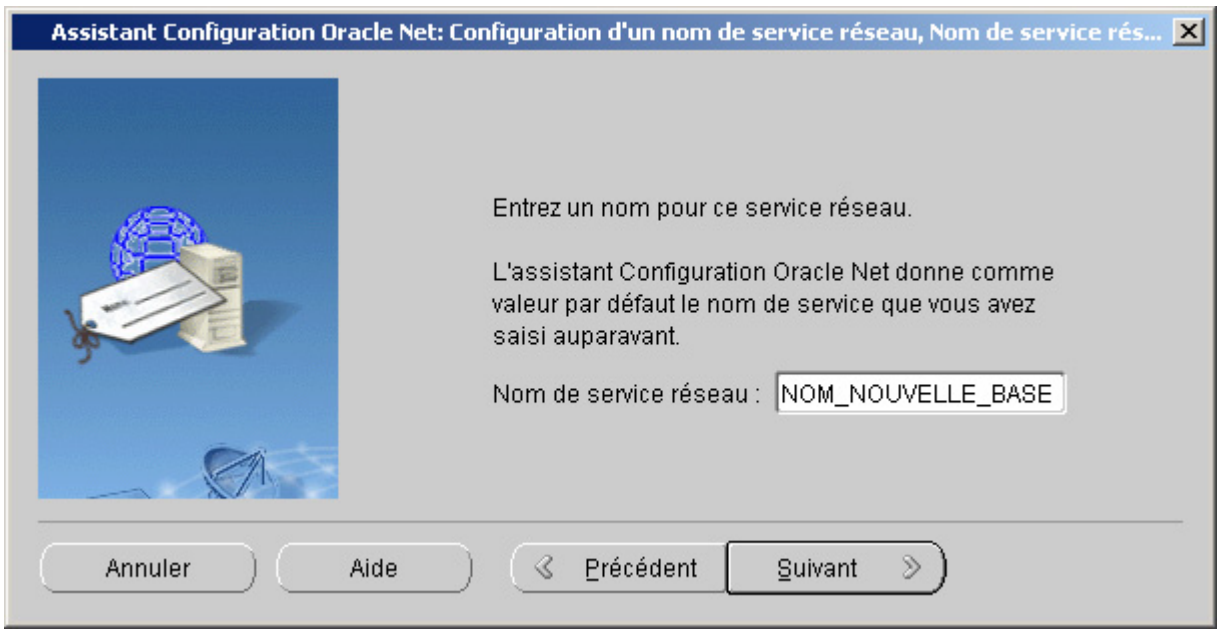
Répondez « Oui » pour effectuer un test de connexion à la nouvelle base, puis Suivant.


Assistant Configuration Oracle Net: Configuration d'un nom de service réseau, Connexion
Veuillez patienter. L'assistant Configuration Oracle Net essaie d'établir la connexion à la base de données à l'aide des informations fournies...
Détails :
Connexion en cours...ORA-01017: nom utilisateur/mot de passe non valide ; connexion refusée
Le test a échoué.
Certaines des informations fournies peuvent être erronées. Cliquez sur Précédent pour revoir les informations fournies sur le nom de service réseau, ou cliquez sur Changer de connexion pour modifier le nom utilisateur.
Changer de connexion
Annuler
Aide
Précédent
Suivant

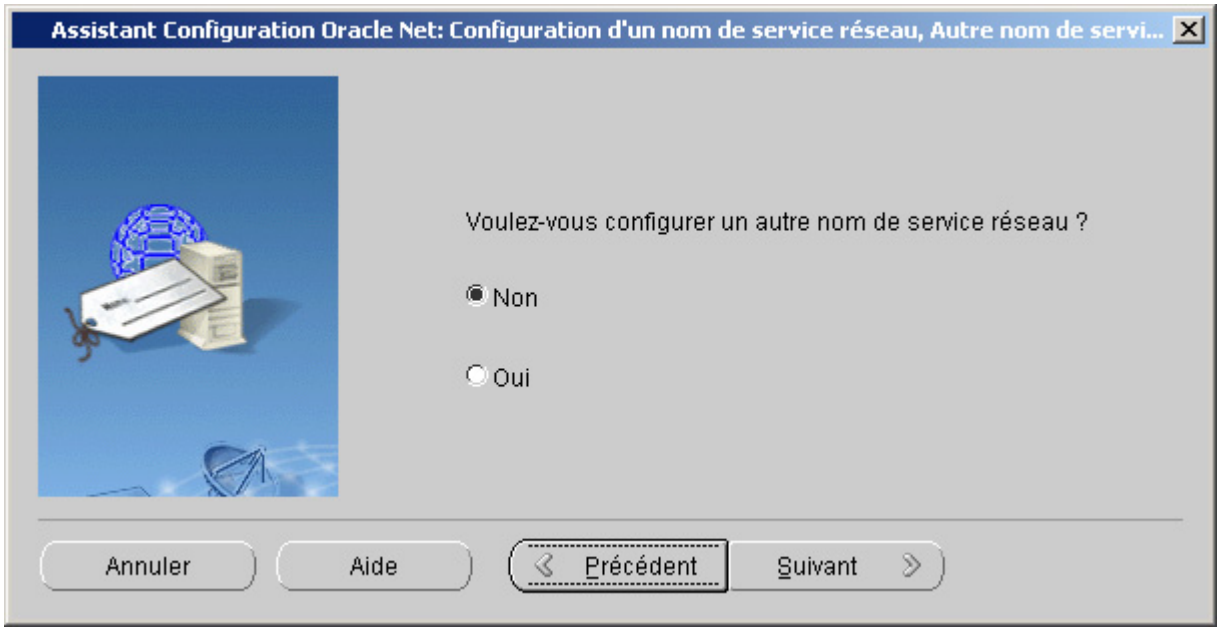
Le test ratera automatiquement la première fois, car le mot de passe est mal configuré. Cliquer sur « Changer de connexion », entrer les informations d'un compte existant sur la base (habituellement le compte system) et recommencer le test.


Assistant Configuration Oracle Net: Configuration d'un nom de service réseau, Connexion
Veuillez patienter. L'assistant Configuration Oracle Net essaie d'établir la connexion à la base de données à l'aide des informations fournies...
Détails :
Connexion en cours...Le test a réussi.
Changer de connexion
Annuler
Aide
Précédent
Suivant

Une fois ceci affiché, la connexion est opérationnelle depuis le client. Faire Suivant.


Assistant Configuration Oracle Net: Configuration d'un nom de service réseau, Nom de service rés...
Entrez un nom pour ce service réseau.
L'assistant Configuration Oracle Net donne comme valeur par défaut le nom de service que vous avez saisi auparavant.
Nom de service réseau :
NOM_NOUVELLE_BASE
Annuler
Aide
Précédent
Suivant

Dans « Nom de service réseau », entrer le nom de la base pour laquelle on configure la connexion, puis Suivant.



Répondre « Non » si il n'y a pas d'autres bases auxquelles on a besoin de se connecter depuis le client, puis Suivant.

La configuration de la connexion est alors terminée.

IV . Fichiers et répertoires importants

IV.1 /ora01/app/oracle/product/10.2.0/db_1/bin/

Répertoire contenant la plupart des exécutables Oracle. Parmi ceux-ci, on retiendra les suivants :

- ***dbshut*** : permet d'arrêter toutes les bases du serveur
- ***dbstart*** : permet de démarrer toutes les bases du serveur
- ***emctl {start | stop }*** : permet l'arrêt / lancement de Enterprise Manager
- ***lsnrctl { start | stop }*** : permet l'arrêt / lancement du listener
- ***isqlplusctl { start | stop }*** : permet l'arrêt / lancement de l'utilitaire isqlplus
- ***tnsping*** : permet de tester la communication avec le listener
- ***sqlplus*** : permet de lancer le programme SQL*Plus.
- ***dbca*** : se lance par terminal ouvert sur l'interface graphique sous login oracle, permet de lancer l'assistant de configuration de base de données : création d'une base, configuration d'une base existante etc…
- ***netca*** : se lance par terminal ouvert sur l'interface graphique sous login oracle, permet de lancer l'assistant de configuration Oracle Net.

IV.2 /ora01/app/oracle/product/10.2.0/db_1/

Il s'agit en fait du répertoire $ORACLE_HOME, dans lequel se trouvent trois fichiers log utiles :

- ***startora.log*** : créé par le script /etc/init.d/ora à chaque fois qu'il est appelé avec le paramètre start, comme au démarrage du serveur. Il regroupe les informations de démarrage de chaque produit Oracle lancés automatiquement par le script /etc/init.d/ora
- ***stoporacle.log*** : créé par le script /etc/init.d/ora à chaque fois qu'il est appelé avec le paramètre stop. Ce fichier log n'est incrémenté que lorsque le script /etc/init.d/ora est appelé explicitement par la commande ora stop. Il n'est pas renseigné lors de l'arrêt du serveur.
- ***startup.log*** : contient les informations concernant le lancement des instances. Est renseigné dès qu'une instance est lancée automatiquement par dbstart.
- ***shutdown.log :*** contient les informations concernant l'arrêt des instances. Est renseigné dès qu'une instance est arrêtée automatiquement par dbshut.

IV.3 /ora01/app/oracle/product/10.2.0/db_1/network/admin/

Ce répertoire contient les fichiers de configuration du listener, de sqlnet et des services réseau

- ***sqlnet.ora*** : définit les méthodes de résolutions de noms lors des connexions à Oracle.
- ***listener.ora*** : c'est ici que le ou les listener disponibles sur le serveur sont paramétrés, avec leur numéro de port, l'hôte sur lequel ils sont disponibles, le protocole de communication utilisé pour leur communication…
- ***tnsnames.ora*** : configuration des services réseau, c'est à dire la correspondance entre les noms des instances et les noms que l'on utilisera pour s'y connecter.

IV.4 /ora01/app/oracle/admin/INSTANCE/bdump/

Répertoire contenant différents fichiers de logs, notamment le fichier alert_INSTANCE.log, retraçant tous les incidents que rencontre la base.

IV.5 /ora01/app/oracle/oradata/INSTANCE/

Répertoire contenant les fichiers de données (*.dbf), les fichiers de contrôles (*.ctl) et les fichiers de journalisation (redologs : redo0x.log). On notera qu'il existe en théorie un fichier de données par tablespace créé sur la base.

IV.6 /etc/oratab

Fichier contenant la liste des bases hébergées sur le serveur. Chaque création d'une nouvelle base par l'assistant configuration de base de données mettra automatiquement à jour ce fichier. Cependant, une création de base manuelle obligera à éditer ce fichier à la main. Les lignes qui le composent sont de la forme :

```
NOM_BASE:REPERTOIRE_ORACLE_HOME:AUTO
```

Le paramètre AUTO permet au script dbstart de savoir s'il doit ou non lancer cette base lorsqu'il est appelé. Ce paramètre est soit Y, soit N. Nous avons donc un fichier oratab formaté comme ci-dessous :

```
ORATEST:/ora01/app/oracle/product/10.2.0/db_1/:Y
```

Si l'on crée la base BASE2 manuellement sur le serveur, il faudra ajouter la ligne suivante dans /etc/oratab :

```
BASE2:/ora01/app/oracle/product/10.2.0/db_1:Y
```

IV.7 /ora01/app/oracle/product/10.2.0/db_1/dbs/

Répertoire contenant les fichiers d'initialisation des instances hébergées sur le serveur.

- ***spfileORACLE_SID.ora*** : fichier d'initialisation d'instance, utilisé par Oracle lors du lancement de l'instance en question.
- ***init.ora*** : fichier d'initialisation global, utilisé en cas d'absence du fichier spfileORACLE_SID.ora.

Attention toutefois à ne pas modifier un fichier spfile. Une telle action ne pourrait pas être réalisée sans corrompre le fichier, entraînant l'impossibilité pour l'instance de se lancer.

V . Annexes

Voici le contenu des principaux scripts créés ou édités par mes soins :

V.1 */etc/init.d/ora*

Script lancé au démarrage de la machine et à son extinction. Il lance tous les produits Oracle qui ont été installés sur le serveur, à savoir :

- ***iSQL*Plus*** : outil permettant de passer des requêtes sur la base par interface web
- ***le listener*** : processus d'écoute recevant les demandes de connexions au serveur Oracle puis qui les soumet à ou aux instances concernées pour authentification. Sans lui, toute connexion à une quelconque base est strictement impossible.
- ***la base*** : lancement de l'instance, montage des fichiers de données de la base sur cette instance et ouverture de la base.
- ***Oracle Enterprise Manager*** : interface web de gestion de base de données.

L'exécution de ce script peut rendre le démarrage de la machine un peu plus lent qu'habituellement.

Ce script peut aussi être lancé indépendamment d'un boot de la machine, pour des reprises sur incidents, pour des sauvegardes... Son emplacement ayant été définit dans le PATH de root et d'oracle, il suffit de taper :

```
ora stop
ora start
```

Script /etc/init.d/ora

```
#!/bin/ksh
#
#
#lancement d'oracle
#
ORACLE_HOME=/ora01/app/oracle/product/10.2.0/db_1
ORACLE=oracle
VAR_LOCK=/var/lock/subsys/dbora

if [ ! "$2" = "ORA" ] ; then
su - $ORACLE -c "$0 $1 ORA"
exit
fi

PATH=${PATH}:$ORACLE_HOME/bin
export ORACLE_HOME PATH

case $1 in
'start')

#Cette partie est exécutée lorsque l'on entre en runlevel 3 ou 5. Le système sait qu'il doit passer le paramètre start au script grâce au nom du lien symbolique dans /etc/rc.d/rc3.d/ et rc5.d/

LOG=$ORACLE_HOME/startoracle.log
touch $LOG
chmod a+r $LOG
echo "********************************" >> $LOG
echo "* $0 : demarrage *" >> $LOG
echo "********************************" >> $LOG
date >> $LOG
echo "---------------------" >> $LOG
echo "demarrage du listener" >> $LOG
echo "---------------------" >> $LOG
$ORACLE_HOME/bin/lsnrctl start >> $LOG 2>&1
echo "--------------------" >> $LOG
echo "demarrage des bases" >> $LOG
echo "--------------------" >> $LOG
$ORACLE_HOME/bin/dbstart >> $LOG 2>&1
echo "---------------------" >> $LOG
echo "demarrage de iSQLplus" >> $LOG
echo "---------------------" >> $LOG
$ORACLE_HOME/bin/isqlplusctl start >> $LOG 2>&1
echo "------------------------------" >> $LOG
echo "demarrage de entreprise manager" >> $LOG
echo "------------------------------" >> $LOG

#Obligation d'utiliser une boucle car il existe une console Enterprise Manager par instance. Il faut donc lancer la console pour chaque instance, en se basant sur la liste des bases /etc/oratab. La console Enterprise Manager se base sur la valeur de la variable ORACLE_SID pour savoir sur quelle instance elle doit se lancer.

i=1
nbline=`cat /etc/oratab | grep -v "#" | sed 1d | wc -l`
while [ "$i" -le "$nbline" ]
do
cat -n /etc/oratab | grep -v "#" | sed 1d | awk -F: '{print $1}' > fictmp
ORACLE_SID=`cat fictmp | grep $i | awk -F" " '{print $2}'`
export ORACLE_SID
$ORACLE_HOME/bin/emctl start dbconsole >> $LOG 2>&1 &
i=`expr $i + 1`
done
rm -f fictmp
;;
```

```
'stop')

#Bloc exécuté lorsque l'on passe en runlevel 0 ou 6. Exécution et passage de paramètres selon le même principe que pour le start, avec utilisation des noms des liens symboliques dans /etc/rc.d/rc0.d/ et rc6.d/

LOG=$ORACLE_HOME/stoporacle.log
touch $LOG
chmod a+r $LOG
echo "**************************" >> $LOG
echo "* $0 : arret *" >> $LOG
echo "**************************" >> $LOG
date >> $LOG
echo "--------------------------" >> $LOG
echo "arret de enterprise manager" >> $LOG
echo "--------------------------" >> $LOG

#Même principe de boucle pour l'arrêt de OEM que pour son démarrage, il faut l'arrêter pour chaque instance présente sur le serveur.

i=1
nbline=`cat /etc/oratab | grep -v "#" | sed 1d | wc -l`
while [ "$i" -le "$nbline" ]
do
cat -n /etc/oratab | grep -v "#" | sed 1d | awk -F: '{print $1}' > fictmp
ORACLE_SID=`cat fic | grep $i | awk -F" " '{print $2}'`
export ORACLE_SID
$ORACLE_HOME/bin/emctl stop dbconsole >> $LOG 2>&1
i=`expr $i + 1`
done
rm -f fictmp

echo "--------------" >> $LOG
echo "arret isqlplus" >> $LOG
echo "--------------" >> $LOG
$ORACLE_HOME/bin/isqlplusctl stop >> $LOG 2>&1
echo "---------------" >> $LOG
echo "arret des bases" >> $LOG
echo "---------------" >> $LOG
$ORACLE_HOME/bin/dbshut >> $LOG 2>&1
echo "-----------------" >> $LOG
echo "arret du listener" >> $LOG
echo "-----------------" >> $LOG
$ORACLE_HOME/bin/lsnrctl stop >> $LOG 2>&1
;;
*)
echo "usage: $0 {start|stop}"
;;
esac

exit
```

III.2 */home/root/.bash_profile :*

Sur le serveur, lorsque l'utilisateur root se connecte, il lui est demandé de choisir l'environnement Oracle qu'il veut charger pour valoriser la variable ORACLE_SID. C'est dans le fichier .bash_profile que le menu de choix et le traitement du choix utilisateur est réalisé, en se basant sur le contenu du fichier /etc/oratab, pour savoir quelles bases sont présentes sur le serveur.

/home/root/.bash_profile

```
# .bash_profile
# Get the aliases and functions
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi

# User specific environment and startup programs
clear
ORACLE_BASE=/ora01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/10.2.0/db_1
PATH=$PATH:$HOME/bin:/etc/init.d/:$ORACLE_HOME/bin/

#Positionnement de l'environnement Oracle

echo "Choix de l'environnement"
awk -F: '{print $1}' /etc/oratab | grep -v "#" | sed 1d | cat -n  #Affichage du menu
echo " "

echo "Entrez le nom de l'instance utilisee pour cette session"
INSTANCE=instance
ORACLE_SID=oraclesid

while [ "$ORACLE_SID" != "$INSTANCE" ]  #Début du traitement du fichier /etc/oratab
do
i=1
LSTINST=`cat /etc/oratab | grep -v "#" | sed 1d | awk -F: '{print $1}'`
echo $LSTINST > lstinst
nbline=`cat /etc/oratab | grep -v "#" | sed 1d | wc -l`
read ORACLE_SID
        while [ "$i" -le "$nbline" ]  #Test si le choix utilisateur correspond aux choix proposés
        do
        INSTANCE=`cut -d" " -f $i lstinst`
        if [ "$ORACLE_SID" = "$INSTANCE" ]
        then
        echo "Positionnement de l'environnement "$ORACLE_SID
        export ORACLE_SID
        break
        else
        i=`expr $i + 1`
        fi
        done
if [ "$ORACLE_SID" = "$INSTANCE" ] #Deuxième test pour affichage d'erreur
then
break
else
echo "Choix incorrect, veuillez recommencer"
fi
done

export PATH ORACLE_BASE ORACLE_HOME
unset USERNAME   #Dévalorisation des variables temporaires du script
unset INSTANCE
unset LSTINST
unset nbline
```